Student ID:

Serial Number:

Suppose a binary tree T holds keys $k_1, k_2, \ldots, k_n$. Let w_i denote a weight that is assigned to k_i. A large weight means that k_i is accessed more often, and vice versa. Now, if d_i denotes 1+ the depth of k_i (e.g. the root has d_i of 1), then the weighted access cost of T is defined as $wac(T) = \sum_{i=1}^{n} w_i \times d_i$

Thus, the problem to be solved is to find a binary search tree that holds keys $k_1, k_2, \ldots, k_n$, and has minimal weighted access cost. Let $wac(i, j)$ denote the minimum attainable weighted access cost for any binary search tree that holds keys $i, i+1, \ldots, j$. Then the dynamic-programming recurrence needed to solve the Optimal Binary Search Tree problem is as follows:

$$wac(i,j) = \begin{cases} 0, & i > j \\ w_i, & i = j \\ \min\limits_{i \le k \le j} (wac(i, k-1) + wac(k+1, j)) + \sum\limits_{r=i}^{j} w_r, & \text{otherwise} \end{cases}$$

Question 1.1 [0.5 Points]

Suppose keys 1-4 have respective weights 30,60,10,80, and are inserted into an initially empty binary search tree T in the order 4,2,1,3. Determine wac(T).

4

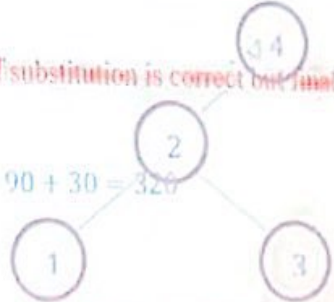
k_i	w_i
1	30

Question 1 [/0.5 Points]

Suppose keys 1-4 have respective weights 30,60,10,80, and are inserted into an initially empty binary search tree T in the order 4,2,1,3. Determine wac(T).

Note: you don't need dynamic programming for this question. //deduct 0.1 if substitution is correct but final answer is not.

$$wac(T) = 80x1 + 60x2 + 30x3 + 10x3 = 80 + 120 + 90 + 30 = 320$$



k_i	w_i
1	30
2	60
3	10
4	80

Question 2 [/ 4.2 Point]

2.1. Use dynamic programming (by completing the tables below) to determine the binary search tree of

$$wac(1,2)=\min\begin{cases} k=1: wac(1,0)+wac(2,2)+30+60=0+60+90=150 \\ k=2: wac(1,1)+wac(3,2)+30+60=30+0+90=120 \end{cases}=120$$

$$wac(2,3)=\min\begin{cases} k=2: wac(2,1)+wac(3,3)+60+10=0+10+70=80 \\ k=3: wac(2,2)+wac(4,3)+60+10=60+0+70=130 \end{cases}=80$$

$$wac(3,4)=\min\begin{cases} k=3: wac(3,2)+wac(4,4)+10+80=0+80+90=170 \\ k=4: wac(3,3)+wac(5,4)+10+80=10+0+90=100 \end{cases}=100$$

$$wac(1,3)=\min\begin{cases} k=1: wac(1,0)+wac(2,3)+30+60+10=0+80+100=180 \\ k=2: wac(1,1)+wac(3,3)+30+60+10=30+10+100=140 \\ k=3: wac(1,2)+wac(4,3)+30+60+10=120+0+100=220 \end{cases}=140$$

$$\begin{cases} k=2: wac(2,1)+wac(3,4)+60+10+80=0+100+150=250 \end{cases}$$

$$\ldots wac(4,4) + 10 + 80 = 0 + 80 + 90 = 170$$

$$\Big(k = 4: wac(3,3) + wac(5,4) + 10 + 80 = 10 + 0 + 90 = 100 \quad = 100$$

$$wac(1,3) = \min \begin{cases} k = 1: wac(1,0) + wac(2,3) + 30 + 60 + 10 = 0 + 80 + 100 = 180 \\ k = 2: wac(1,1) + wac(3,3) + 30 + 60 + 10 = 30 + 10 + 100 = 140 \\ k = 3: wac(1,2) + wac(4,3) + 30 + 60 + 10 = 120 + 0 + 100 = 220 \end{cases} = 140$$

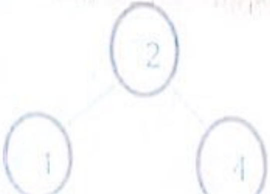
$$wac(2,4) = \min \begin{cases} k = 2: wac(2,1) + wac(3,4) + 60 + 10 + 80 = 0 + 100 + 150 = 250 \\ k = 3: wac(2,2) + wac(4,4) + 60 + 10 + 80 = 60 + 80 + 150 = 290 \\ k = 4: wac(2,3) + wac(5,4) + 60 + 10 + 80 = 80 + 0 + 150 = 230 \end{cases} = 230$$

$$\Big(k = 1: wac(1,0) + wac(2,4) + 30 + 60 + 10 + 80 = 0 + 230 + 180 = 410$$

$\ldots 0 + 150 = 230$

$$wac(1,4) = \min\begin{cases} k = 1: wac(1,0) + wac(2,4) + 30 + 60 + 10 + 80 = 0 + 230 + 180 = 410 \\ k = 2: wac(1,1) + wac(3,4) + 30 + 60 + 10 + 80 = 30 + 100 + 180 = 310 \\ k = 3: wac(1,2) + wac(4,4) + 30 + 60 + 10 + 80 = 120 + 80 + 180 = 380 \\ k = 4: wac(1,3) + wac(5,4) + 30 + 60 + 10 + 80 = 140 + 0 + 180 = 320 \end{cases} = 310$$

2.2. Draw the minimum cost tree resulting from your answer in Question 2.1. //0.3 pt: consider root table: 0.1 root, 0.1 root of left subtree and 0.1 root of right subtree. If a student draws node 3 as the root of the left subtree and under it node 4 then we cut 0.1 mark.

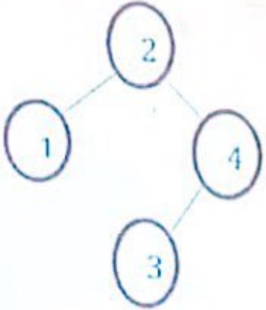


	0	1	2	3	4
1	0	30	120	140	310
2		0	60	80	230
3			0	10	100
4				0	80
5					0

	0	1	2	3	4
1		1	2	2	2
2			2	2	4
3				3	4
4					4
5					

//We will only consider final values in main table-not recurrences. 12 value to compute in main table-each 0.3 marks

... root of right subtree. If a student draws node 3 as the root of the left subtree and under it node 4 then we cut 0.1 mark.



2.3. What is the weighted access cost of the optimal binary search tree for the instance in Question 2.1?

310 //0.3 pt: consider top-right value in main table

Question 3 [/ 0.3 Point]

What is the computational complexity of the dynamic programming method to determine ...

Question 3 [/ 0.3 Point]

What is the computational complexity of the dynamic programming method to determine the binary search tree of minimum weighted-access cost? $O(n^3)$ or $O(n^2)$